Tracking Across Multiple Cameras With Disjoint Views

Omar Javed Zeeshan Rasheed Khurram Shafique Mubarak Shah
Computer Vision Lab
University of Central Florida
{ojaved,zrasheed,khurram,shah}@cs.ucf.edu

Abstract

Conventional tracking approaches assume proximity in space, time and appearance of objects in successive observations. However, observations of objects are often widely separated in time and space when viewed from multiple non-overlapping cameras. To address this problem, we present a novel approach for establishing object correspondence across non-overlapping cameras. Our multi-camera tracking algorithm exploits the redundance in paths that people and cars tend to follow, e.g. roads, walk-ways or corridors, by using motion trends and appearance of objects, to establish correspondence. Our system does not require any inter-camera calibration, instead the system learns the camera topology and path probabilities of objects using Parzen windows, during a training phase. Once the training is complete, correspondences are assigned using the maximum a posteriori (MAP) estimation framework. The learned parameters are updated with changing trajectory patterns. Experiments with real world videos are reported, which validate the proposed approach.

1. Introduction

Surveillance of wide areas requires a network of cameras. It is not always possible to have overlapping camera views in this case. The observations of the same object can be widely separated in time and space in such a scenario. Moreover, it is preferable that the tracking system does not require camera calibration or complete site modelling, since the luxury of calibrated cameras or site models is not available in most situations. In this paper, we focus on the problem of multi-camera tracking in a system of non-overlapping uncalibrated cameras. The task of a multi-camera tracker is to establish correspondence between observations of objects across cameras. We assume that tracking information is available for individual cameras, and the objective is to find correspondences between these tracks, in different cameras, such that the corresponded tracks belong to the same object in the real world.

We use the observations of people through the system of cameras to discover the relationships between the cameras. For example, suppose two cameras A and B are successively arranged alongside a walkway. Suppose people moving along one direction of the walkway that are initially observed in camera A are also observed entering camera B after a certain time interval. However, people moving in opposite direction in camera A might not later be observed in camera B. Thus, the usual locations of exits and entrances between cameras, direction of movement and the average time taken to reach from A to B can be used to constrain correspondences. In this paper, we refer to these cues as *space-time* cues. Another cue for tracking is the appearance of persons as they move through cameras. We present a MAP estimation framework to use these cues in a principled manner for tracking. We use Parzen windows, also known as kernel density estimators, to estimate the inter-camera space-time probabilities from the training data, i.e., probability of an object entering a certain camera at a certain time given the location, time and velocity of its exit from other cameras. Using Parzen windows lets the data 'speak for itself' ([13]) rather than imposing assumptions. The change in appearance as a person moves between certain cameras is modelled using the distances between color models. The correspondence probability, i.e. the probability that two observations are of the same object, depends on both the space-time information and the appearance. Tracks are assigned by estimating the correspondences, which maximize the posterior probabilities. This is achieved by transforming the MAP estimation problem into a problem of finding the path cover of a directed graph for which an efficient optimal solution exists.

The paper is organized as follows: We give an overview of the related research in Section 2. A Bayesian formulation of the problem is presented in Section 3. The learning of path and appearance probabilities is discussed in Section 4. A method to find correspondences that maximizes the a posteriori probabilities is given in Section 5. The procedure to update the probabilistic models is given in Section 6. Results are presented in Section 7.

2. Related Work

A large amount of work on multi-camera surveillance assumes overlapping views. Jain and Wakimoto [9] used cal-

ibrated cameras and an environmental model to obtain 3D location of a person. Cai and Aggarwal [1], used multiple calibrated cameras for surveillance. Geometric and intensity features were used to track the objects in multiple views. Collins et al. [3] developed a system consisting of multiple calibrated cameras and a site model. They used region correlation and location on the 3D site model for tracking. Bayesian Networks were used by (Chang and Gong) [2] and (Dockstader and Tekalp)[5] for tracking and occlusion reasoning across cameras with overlapping views. Lee et al. [14] proposed an approach for tracking in cameras with overlapping FOV's that does not require calibration. The camera calibration information was recovered by matching motion trajectories obtained from different views and plane homographies were computed from the most frequent matches. Khan et al. [12] used field of view (FOV) line constraints for tracking in cameras with overlapping views. Javed et al. [10] extended this approach for tracking in non-overlapping cameras.

Our work is inspired by the approaches of Huang and Russel [8], and Kettnaker and Zabih [11] for tracking in non-overlapping cameras. Huang and Russel presented a probabilistic approach for object identification across two cameras. Their task was to correspond vehicles across cameras on a highway. The appearance of vehicles was modelled by the mean of the color. Transition times were modelled as Gaussian distributions and it was assumed that the initial transition probabilities were known. Our work is different from the above mentioned approach in that, Huang and Russel presented an application specific solution, i.e., tracking vehicles across two calibrated cameras, where vehicles are moving only in one direction and only on specified lanes. We present a general solution, which allows movement in all direction for arbitrary number of un-calibrated cameras. We do not assume that the transition probabilities are known. In addition, their online correspondence algorithm trades off matching confidence with solution space coverage, which forces them to commit early and possibly make an erroneous correspondence. Moreover, they modelled appearance by just the mean color value of the whole object, which is not enough to distinguish between multi-colored objects like people.

Kettnaker and Zabih [11] used a Bayesian formulation of the problem to reconstruct the paths of objects across multiple cameras. The problem was transformed into a linear program to establish correspondence. They required *manual* input of the topology of allowable paths of movement and the transition probabilities. It was also assumed that the paths and transition probabilities were constant. Thus their approach would not be able to cope with any change in the assumed paths of people. We automatically learn the relationship between cameras together with the most likely path probabilities and transition time intervals. Moreover, we update the probabilities online to keep up with changing traffic patterns. In their formulation, they used assumptions different from ours, which lead to different correspondence probabilities. Furthermore, they did not jointly model the positions, velocities and transition times of objects across cameras. We do not assume independence between these correlated features.

3 Formulation

Suppose that we have a system of k Cameras $C_1, C_2, \ldots, C_k$ with non-overlapping views. Further, assume that there are n objects $p_1, p_2, \ldots, p_n$ in the environment, such that each object p_i generates a sequence of tracks $T_i = T_{i,t_1}, T_{i,t_2}, \ldots, T_{i,t_m}$ in the system of cameras at successive time instances $\{t_m\}$. Assuming that the task of single camera tracking is already solved, let $O_j = \{O_{j,1}, O_{j,2}, \ldots, O_{j,m_j}\}$ be the set of observations (tracks) that were observed by the camera C_j. We assume each observation $O_{j,a}$ to be based on two features, appearance of the object $O_{j,a}(app)$ and space-time features of the object $O_{j,a}(st)$ (location, velocity, time etc.). It is reasonable to assume that both $O_{j,a}(app)$ and $O_{j,a}(st)$ are independent of each other, i.e., the appearance of an object doesn't depend on its space-time cues in the image. Let a correspondence $k_{a,b}^{c,d}$ to be an ordered pair $(O_{a,b}, O_{c,d})$, which defines the hypothesis that the observations $O_{a,b}$ and $O_{c,d}$ correspond to the consecutive tracks of the same object in the environment. The problem of Multi-camera tracking is to find a set of correspondences $K = \left\{k_{a,b}^{c,d}\right\}$ such that

- For all $k_{a,b}^{c,d}, k_{p,q}^{r,s} \in K$, $k_{a,b}^{c,d} \neq k_{p,q}^{r,s} \Rightarrow (a,b) \neq (p,q) \wedge (c,d) \neq (r,s)$, i.e. each observation of an object is preceded or succeeded by a maximum of one observation.
- $k_{a,b}^{c,d} \in K$ if and only if $O_{a,b}$ and $O_{c,d}$ correspond to the consecutive tracks of the same object in the environment.

Now, let $K = \left\{k_{i,a}^{j,b}\right\}$ be a hypothesized solution of the above problem. Assuming that each correspondence, i.e. a matching between two observations, is independent of other observations and correspondences, we have,

$$P(K|O) = \prod_{k_{i,a}^{j,b} \in K} P_{i,j}\left(k_{i,a}^{j,b} | O_{i,a}, O_{j,b}\right), \quad (1)$$

where $P_{i,j}\left(k_{i,a}^{j,b} | O_{i,a}, O_{j,b}\right)$ is the conditional probability of the correspondence $k_{i,a}^{j,b}$, given the observations $O_{i,a}$ and

$O_{j,b}$ for two cameras C_i and C_j in the system. From Bayes Theorem, we have,

$$P_{i,j}\left(k_{i,a}^{j,b}|O_{i,a},O_{j,b}\right)=\frac{P_{i,j}\left(O_{i,a},O_{j,b}|k_{i,a}^{j,b}\right)P_{i,j}\left(k_{i,a}^{j,b}\right)}{P_{i,j}\left(O_{i,a},O_{j,b}\right)}. \quad (2)$$

Using the above equation along with the independence of observations $O_{j,a}(app)$ and $O_{j,a}(st)$ (for all a and j), we have,

$$\begin{aligned}P\left(K|O\right) \quad = \quad & \prod_{k_{i,a}^{j,b}\in K}((\frac{1}{P_{i,j}\left(O_{i,a},O_{j,b}\right)}) \\ & P_{i,j}\left(O_{i,a}(app),O_{j,b}(app)|k_{i,a}^{j,b}\right) \quad (3)\\ & P_{i,j}\left(O_{i,a}(st),O_{j,b}(st)|k_{i,a}^{j,b}\right)P_{i,j}\left(k_{i,a}^{j,b}\right)).\end{aligned}$$

The solution of the multi-camera tracking problem is the hypothesis K' in the hypothesis space Σ that maximizes the above term (posterior) and is given by

$$K'=\arg\max_{K\in\Sigma}P\left(K|O\right).$$

We define the prior $P_{i,j}\left(k_{i,a}^{j,b}\right)$ to be the probability $P\left(C_i,C_j\right)$ of a transition from camera C_i to C_j. Moreover, we assume that the observation pairs are uniformly distributed and hence, $P_{i,j}\left(O_{i,a},O_{j,b}\right)$ is a constant scale factor. Thus the problem is reduced to the solution of following term:

$$\begin{aligned}K' \quad = \quad & \arg\max_{K\in\Sigma}\sum_{k_{i,a}^{j,b}\in K}\log(P_{i,j}\left(O_{i,a}(app),O_{j,b}(app)|k_{i,a}^{j,b}\right) \\ & P_{i,j}\left(O_{i,a}(st),O_{j,b}(st)|k_{i,a}^{j,b}\right)P\left(C_i,C_j\right)). \quad (4)\end{aligned}$$

In order to maximize the posterior, we need to find the space-time and appearance probability density functions. This issue is discussed in the next section.

4 Learning Inter-Camera Space-Time and Appearance Probabilities

Learning is carried out by assuming that the correspondences are known. One way to achieve this is by making a single person roam in the environment. However, it is not always possible to have a single person in the environment. In this case, only appearance matching can be used for establishing correspondence, since path information is unknown. Note that, during training, it is not necessary to correspond all persons across cameras. Only the best matches (those closest in appearance) can be used for learning.

4.1 Estimating inter-camera space-time probabilities using Parzen windows

The Parzen window technique is used to estimate the space-time pdfs between each pair of cameras. Suppose we have a sample S consisting of n, d dimensional, data points $x_1,x_2,\ldots,x_n$ from a multi-variate distribution $p(x)$, then an estimate $\hat{p}(x)$ of the density at x can be calculated using

$$\hat{p}(x)=\frac{1}{n}|H|^{-\frac{1}{2}}\sum_{i=1}^{n}K(H^{-\frac{1}{2}}(x-x_i)), \quad (5)$$

where the d variate kernel $K(x)$ is a bounded function satisfying $\int K(x)dx=1$, and H is a symmetric $d\times d$ bandwidth matrix. The multivariate kernel $K(x)$ can be generated from the product of symmetric univariate kernel K_u, i.e.

$$K(x)=\prod_{j=1}^{d}K_u(x_{\{j\}}). \quad (6)$$

The feature vector x, used for learning the space-time pdfs from camera C_i to C_j, i.e., $P_{i,j}(O_{i,a}(st),O_{j,b}(st)|k_{i,a}^{j,b})$, is a seven dimensional vector, consisting of the exit location from C_i, entry locations in C_j, exit velocities, and the time taken between exit and entry. We use a univariate Gaussian kernel to generate $K(x)$. Moreover, to reduce the complexity, H is assumed to be a diagonal matrix, i.e., $H=diag[h_1^2,h_2^2,\ldots,h_d^2]$. Each time, a correspondence is made during the training phase, the observed feature is added to the sample S.

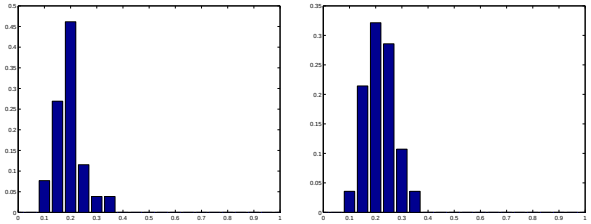
Note that the events, of an object exiting from one camera and entering into another, will be separated by a certain time interval. We refer to this interval as *inter-camera travel time*. Following are some key observations that we have modelled in our system.

- The inter-camera travel time is dependent on the magnitude and direction of motion of the person.
- The inter-camera travel time is also dependent on the location of exit from one camera and the location of entrance in the other.
- The locations of exits and entrances across the cameras are also correlated.

Since the correspondences are known in the training phase, the likely time intervals and the exit/entrance locations are learned by estimating the pdf. The reason for using the Parzen window approach for estimation is that, rather than imposing assumptions, the nonparametric technique allows us to directly approximate the d dimensional density describing the joint pdf. It is also guaranteed to converge to any density function with enough training samples

[6]. Moreover, it does not impose any restrictions on the shape of the function, neither it assumes independence between the feature set.

The prior probability of correspondence of an object moving from C_i to C_j, i.e. $P(C_i, C_j)$, is calculated from the ratio of people that exit C_i and enter C_j to the total number of people that exited C_i, during the learning phase. The priors are normalized to sum up to one.



(a) (b)

Figure 1: Histograms of modified Bhattacharya distance for correspondences obtained during a training sequence. Histograms are shown when a person enters (a) camera 2 from camera 1 (see Figure 2) (b) camera 1 from camera 2.

4.2 Estimating change in appearances across cameras

In addition to the position and time information, we want to model the change in appearance of a person from one camera to another. We represent the appearance by color histograms. The idea here is to learn the usual change in color of people as they move between cameras and use this cue for establishing correspondence. The distance D between two m bin histograms k and q is computed as

$$D(k,q) = \sqrt{1 - \sum_{i=1}^{m} \sqrt{\hat{k}_i \hat{q}_i}}. \qquad (7)$$

This is the modified Bhattacharyya coefficient [4]. One advantage of using this distance measure is that it is a metric. Since in the training phase the correspondences are known, the distance D between two observations of the same object can be measured. We fit a Gaussian distribution to these distances that were calculated during the training phase. Figure 1 shows the histograms of distances. Note that the shape is fairly close to that of a Gaussian distribution. Now we define probability $P_{i,j}(O_{i,a}(app), O_{j,b}(app)|k_{i,a}^{j,b})$ equal to:

$$p_{i,j}(D(a,b)) = \frac{1}{\sqrt{2\pi\sigma_{i,j}^2}} e^{-\frac{1}{2}(\frac{D(a,b)-\mu_{i,j}}{\sigma_{i,j}})^2}, \qquad (8)$$

where $\mu_{i,j}$ and $\sigma_{i,j}^2$ are the mean and variance respectively, which are calculated from the training color distance data obtained from correspondences of observations from camera i to camera j. Note that if the cameras have non-linear gain functions then the supposition that all the colors will change similarly is not true.

5 Establishing Correspondences

The problem of finding the hypothesis that maximizes the a posteriori probability can be modelled as finding the path cover of a directed graph as follows: We construct a directed graph such that for each observation $O_{i,a}$, there is a corresponding node in the directed graph, while each hypothesized correspondence $k_{i,a}^{j,b}$ is modelled by an arc from node of observation $O_{i,a}$ to the node of observation $O_{j,b}$. The weight of this arc of hypothesized correspondence $k_{i,a}^{j,b}$ is the term, in the summation in Eq. 4, which relates to $k_{i,a}^{j,b}$. It can easily be seen from the definition of the solution (as given in Section 3), that a hypothesized solution K is a set of disjoint directed paths in the graph, covering the entire graph (i.e., every vertex is in exactly one of the paths in K). The solution of the MAP estimation problem is a hypothesis K, such that the sum of weights of the arcs in K is maximum among all such sets. This problem can be optimally solved in polynomial time, if the directed graph is acyclic, by reducing it to finding the maximum matching of an undirected bipartite graph [15]. This bipartite graph is obtained by splitting every vertex v of the directed graph into two vertices v^- and v^+, such that each arc coming into the vertex v is substituted by an edge incident to vertex v^-, while the vertex v^+ is connected to an edge for every arc going out of the vertex v in the directed graph. The maximum matching of a bipartite graph can be found by an $O(n^{2.5})$ algorithm by Hopcroft and Karp [7], where n is the total number of vertices in graph G, i.e., the total number of observations in the system.

The method defined above, assumes that the entire set of observations is available and hence cannot be used in real time applications. A standard approach to handle this type of problems in real time applications is to use a sliding window of a fixed time interval. This approach is not optimal, and the selection of window size is a tradeoff between the quality of results and the timely availability of the output. In order to avoid early commitment and making an erroneous correspondence, we adaptively select the size of sliding window in the online version of our algorithm. This is achieved by examining the space-time pdfs for all observations (tracks) in the environment that are not currently visible in any of the cameras in the system and finding the time interval after which the probability of reappearance of all these observations in any camera is below certain threshold.

6 Online update of Space-time and Appearance Models

The motion trends of the objects can change with time, so there is a need to update the position and appearance models during the lifetime of the system. The kernel density estimates can be updated simply, by adding the current observation of a correspondence to the sample set. However, we want to 'forget' the obsolete data too. This can be achieved by estimating the density only from the most recent N samples. For real time requirements, an efficient method for online kernel density estimation has been proposed by Lambert et al. [13]. The Gaussian distribution for the change in appearance model is updated by an exponential decay scheme

$$\mu_t = (1-\rho)\mu_{t-1} + \rho D_t, \quad (9)$$

$$\sigma_t^2 = (1-\rho)\sigma_{t-1}^2 + \rho(D-\mu_t)^2, \quad (10)$$

where D is the distance between appearances calculated from the most recent correspondence and ρ is a learning parameter.

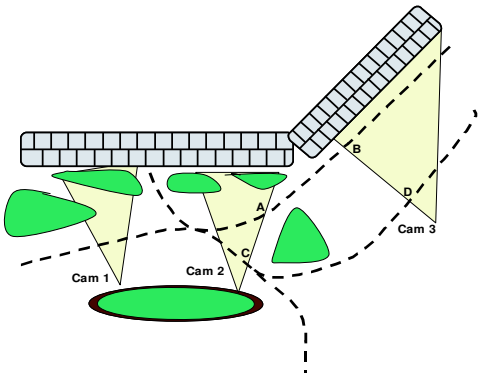
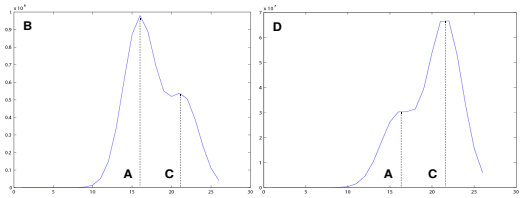


Figure 2: Camera setup for *Business Sequence.* The cameras were mounted approximately 15 to 30 yards apart. Dotted lines are some of the paths a person can take. Field-of-view of each camera is also shown with triangles. It took 8 to 14 seconds for a person walking at normal speed to exit from the view of camera 1 and enter camera 2. The walking time between camera 2 and 3 was 10 to 18 seconds. The dark regions are patches of grass. Most people avoid walking over them.

7 Results

To evaluate the performance of our system, we performed experiments with two different camera settings. In order to estimate bandwidth matrix $H = diag[h_1^2, h_2^2, \ldots, h_d^2]$, the range of each space-time feature for each camera pair was computed during training. The bandwidth of each feature was set to be one tenth of the range. Single camera tracking was performed by the method proposed in [10].



(a) (b)

Figure 3: Marginal pdf w.r.t. position for transitions from camera 2 to 3. A,B,C and D are the points of entry/exit as seen in Fig. 2. (a) The probability of entering camera 3 at point B from camera 2 (the peak corresponds to point A). (b) The probability of entering camera 3 at point D from camera 2 (the peak corresponds to the point C). Note that the points A,B,C and D are given for illustrative purposes. The space-time pdf's map all boundary coordinates of one image to another.

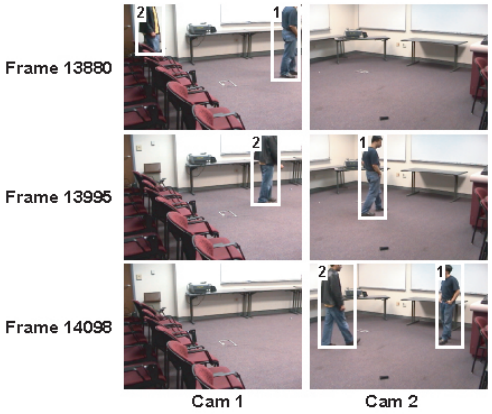


Figure 4: Consistent labelling with two cameras and two people in the scene. Correct Correspondences are obtained as the persons move across the cameras.

The first experiment was done with two cameras in a room. The training phase lasted for ten minutes and the test was run for five minutes. In the testing phase, a total of thirteen tracks were recorded in individual cameras. Our algorithm assigned correct labels for all transitions within the cameras and detected that there were only two persons in the environment. The second experiment was an outdoor sequence and involved three cameras C_1, C_2, C_3. The camera settings and their field of views are shown in Figure 2. Training was done on a ten minute sequence in the presence of multiple persons. Figure 3 shows the probabilities of entering C_3 at point B and D from C_2. Here the space-time pdf correctly demonstrates the fact that the probability of entry at point B, is higher from point A than from point C.

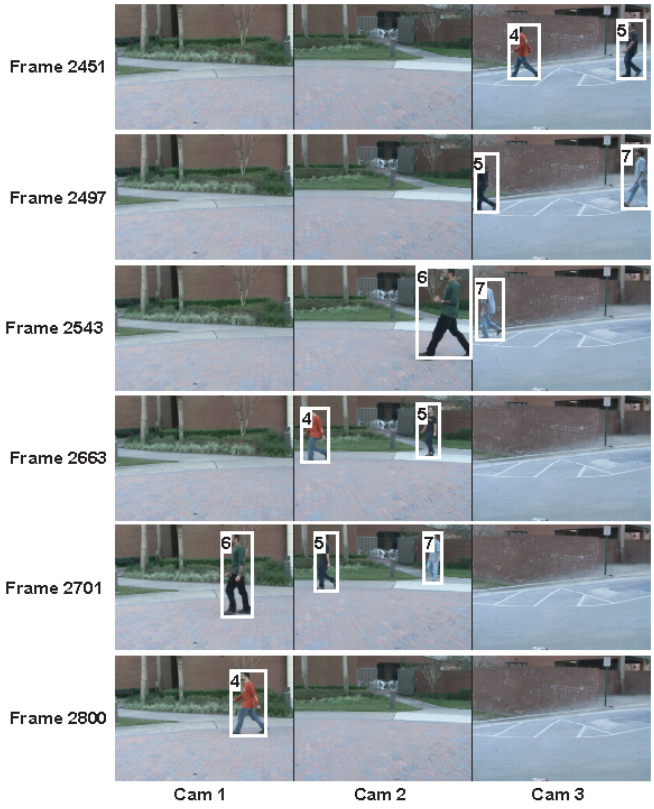


Figure 5: Consistent labelling over time. Row 1 shows person with label 4 moving towards camera 2. In row 2 label 4 is in the unobserved region. However, a person enters camera 2 directly before label 4. This new entry is detected correctly. Correct correspondences are established as people continue to move across the environment, as shown in subsequent frames.

This is because people like to take the shortest possible path between two points. Note that, if a person being observed in C_1 exits from the right side, it is highly unlikely that he would enter C_2 from the right before being observed in other cameras. This fact was also learned by the system and the probability of entering C_2 after exiting from the right side of C_1 was nearly zero. The testing was carried out on a fifteen minute sequence. In this experiment, a total of 71 individual tracks were obtained. Our algorithm detected 45 transitions within the cameras with no error and correctly determined that 27 people walked through the environment. Figures 5 shows some tracking results from the second experiment. One obvious situation, in which the tracking algorithm can assign a wrong label, is if a person takes a much longer time in the unobserved region than expected, e.g., he stands in the unobserved region. Then, if he enters another camera, the time constraint will force the person to be assigned a new label. However, our experiments demonstrate that in most situations tracking is reliable.

8. Conclusions

We have demonstrated that tracking is possible even when observations of objects are not available for relatively large time periods due to non-overlapping camera views. Multiple cues including inter-camera time intervals, location of exit/entrances, and velocities of objects are jointly modelled to constrain correspondences. These cues are combined in a Bayesian framework for tracking. The tracking system begins with some prior knowledge gained from an initial training phase. The learned parameters are continuously updated to keep up with the changing motion and appearance patterns throughout the life-time of the system. We have presented results on realistic scenarios to validate the proposed approach.

References

[1] Q. Cai and J. K. Aggarwal. "Tracking human motion in structured environments using a distributed-camera system". *IEEE Tans. on PAMI*, 2(11), Nov 1999.

[2] T. Chang and S. Gong. "Tracking multiple people with a multi-camera system ". In *IEEE Workshop on Multi-Object Tracking*, 2001.

[3] R. Collins, A. Lipton, H. Fujiyoshi, and T. Kanade. "Algorithms for cooperative multisensor surveillance". *Proceedings of the IEEE*, 89(10), October 2001.

[4] D. Comaniciu, V. Ramesh, and P. Meer. "Real-time tracking of non-rigid objects using mean shift". In *CVPR*, 2000.

[5] S. L. Dockstader and A. M. Tekalp. "Multiple camera fusion for multi-object tracking". In *IEEE Workshop on Multi-Object Tracking*, 2001.

[6] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley, 2000.

[7] J. Hopcroft and R. Karp. "An $n^{2.5}$ algorithm for maximum matchings in bipartite graphs". *SIAM J. Computing*, Dec 1973.

[8] T. Huang and S. Russell. "Object identification in a bayesian context.". In *Proceedings of IJCAI,*, 1997.

[9] R. Jain and K. Wakimoto. "Multiple perspective interactive video". In *IEEE International Conference on Multimedia Computing and Systems*, 1995.

[10] O. Javed, Z. Rasheed, O. Alatas, and M. Shah. "KnightM: A real time surveillance system for multiple overlapping and non-overlapping cameras". In *Proceedings of ICME*, 2003.

[11] V. Kettnaker and R. Zabih. " Bayesian multi-camera surveillance". In *Proceedings of CVPR*, 1999.

[12] S. Khan, O. Javed, Z. Rasheed, and M. Shah. "Human tracking in multiple cameras". In *Proceedings of ICCV*, 2001.

[13] C. G. Lambert, S. E. Harrington, C. R. Harvey, and A. Glodjo. "Efficient on-line nonparametric kernal density estimation". *Algorithmica*, 25, 1999.

[14] L. Lee, R. Romano, and G. Stein. "Monitoring activities from multiple video streams: Establishing a common coordinate frame". *IEEE Trans. on PAMI*, 22(8):758–768, Aug 2000.

[15] K. Shafique and M. Shah. "A non-iterative greedy algorithm for multi-frame point correspondence ". In *Proceedings of ICCV*, 2003.